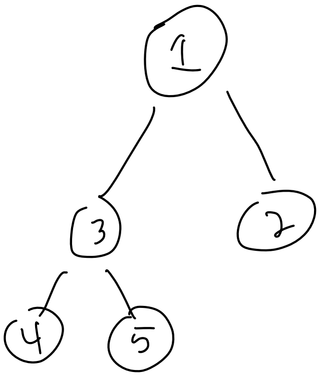
Binary Trees

Proper (full)

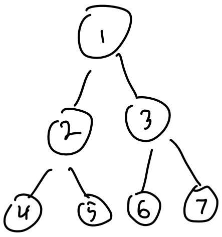
- 0 or 2 children



Perfect

-alway 0 or 2 Children

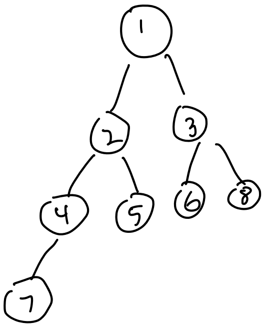
-all leaves at same depth



Complete

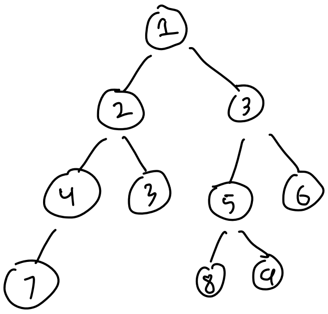
every* level is filled

* except last level

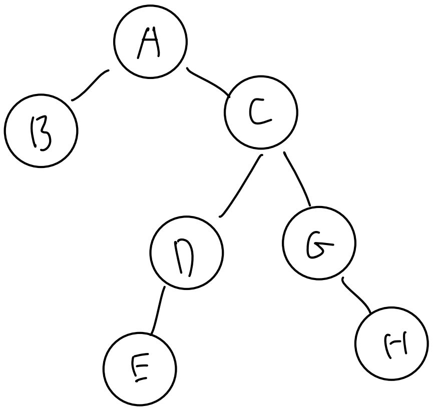


balanced

left and right subtree's height differ by ≤ 1



In order

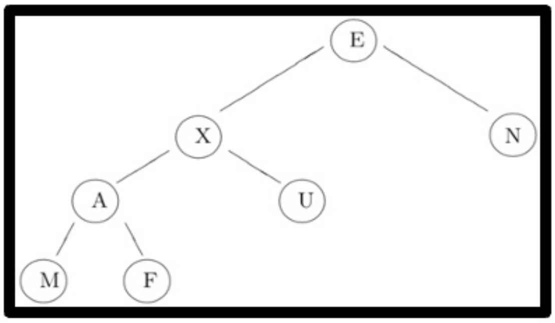


Print: B A E D C H G

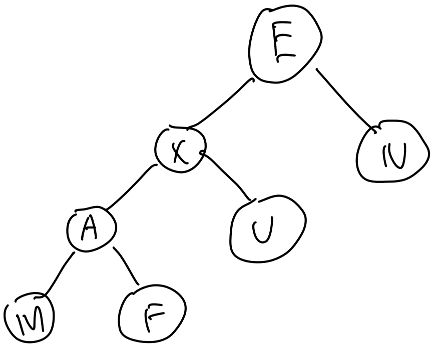
```
void  inOrder(node) {
    if (node == nullptr) {
        return;
    }
        inorder (node → Child1)
        Print (node → val)
        inorder (node → Child2)

  }
```

1. Draw a (single) binary tree T , such that:
   - Each node of T stores a single character
   - A preorder traversal of T yields EXAMFUN
   - An inorder traversal of T yields MAFXUEN
2. What is post order output of the tree given on the right?



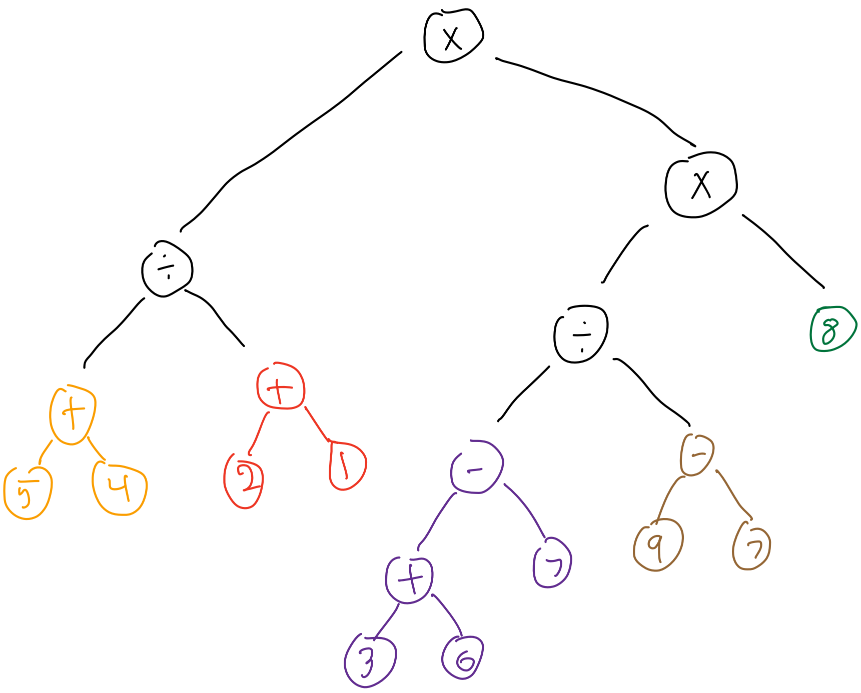
1)



2) MFAUXNE

3)

(((5+ 4)/(2+1))*(((3+6)-7) / (9-7)) * 8)



$$\left((5+4)\div(2+1)\right)\times\left(\left((3+6)-7\right)/(9-7)\right)\cdot 8)$$

4. Draw an arithmetic expression tree that has four external nodes, storing the number 1,3,5, and 8(with each number stored in a distinct external node, but not necessary in this order) and has three internal nodes, each storing an operator from the set {+, -, x, or /}, so that the value of the root is 20. The operators may return and act on fractions, and operator may be used more than once.